Fault detection method for nonlinear systems based on probabilistic neural network filtering

Liu, J.; Scherpen, J.M.A.

Published in:
International Journal of Systems Science

DOI:
10.1080/00207720210000046216

International Journal of Systems Science, 2002, volume 33, number 13, pages 1039–1050

Fault detection method for nonlinear systems based on probabilistic neural network filtering

J. Liu†* and J. M. A. Scherpen‡*

A fault detection method for nonlinear systems, which is based on Probabilistic Neural Network Filtering (PNNF), is presented. PNNF limits the maximum estimation error of the asymptotic Bayes optimal result and describes the tracking process with an expression. On the basis of these properties of PNNF and the statistical characteristics of the noise of the system, a fault threshold can be better calculated, especially for the tracking process corresponding to a strong disturbance. According to the threshold, the fault can be detected by evaluating the residuals. Also, for some special cases when a fault is potential but the system is in steady state, which causes the information for fault detection may be insufficient and a group of disturbances are artificially input with definite amplitudes, so that the result of detection can be enhanced by comparing the real with the expected tracking processes of the filter. Examples are given to demonstrate the method of fault detection based on PNNF.

1. Introduction

Fault detection in control systems plays an important role in industry and presently has considerable interests in more theoretical research. Among the schemes of fault detection, the method based on state estimation is important, and includes two main aspects, residual generation and decision-making (Davs 1974, Willsky 1976, Isermann 1984 and Frank 1987, 1994b). For linear systems, the method of fault detection based on state estimation has been studied extensively. Chow and Willsky (1984) presented a method of robust failure-detection filtering with analytical redundancy. White and Speyer (1987) generated a formulation for the detection filter problem by assignment of the closed-loop eigenstructure under certain constraints. Frank and Ding (1994a) studied a frequency domain approach to optimal robust residual generation. For nonlinear systems, Yu and Shields (1996) have discussed a fault-detection method for a special kind of nonlinear system, i.e. the bilinear systems. Zhang *et al.* (1998) proposed a new approach based on an input–output representation and a local approach to change detection. This representation was obtained through elimination of unknown variables, and the local approach transfers general fault detection and diagnosis problems into an asymptotically equivalent simple problem. Guan and Saif (1990) presented a design method of an unknown input observer for single output nonlinear systems. The traditional method used to estimate the states of nonlinear systems is the well-known Extended Kalman Filter (EKF) (Kalman and Bury 1961, Schmit 1970). There are some obvious disadvantages to this filter, such as its initial value sensitivity and the limit in its tracking ability.

In recent years, with the rapidly growing application of computer technology, the field of neural networks has developed considerably. Vemuri and Polycarpou (1997) investigated a neural-network-based learning methodology with guaranteed robustness and stability properties for the detection of faults in robotic systems with modelling uncertainties, and developed a post-fault robotic system model. Trunov and Polycarpou (2000) described a model-based fault detection scheme for the detection and approximation of incipient and abrupt state and sensor faults in nonlinear systems. Probability neural networks were proposed by Specht

Received 25 August 2000. Revised 22 July 2002. Accepted 20 September 2002.

† FMRIB, University of Oxford, John Radcliffe Hospital, Headington, Oxford OX3 9DU, UK.

‡ Faculty of Information Technology and Systems, Department of Electrical Engineering, Delft University of Technology, PO Box 5031, 2600 GA Delft, The Netherlands.

* To whom correspondence should be addressed. e-mail: Liu@tmrib.ox.ac.uk or J.M.A.Scherpen@its.tudelft.nl

International Journal of Systems Science ISSN 0020–7721 print/ISSN 1464–5319 online © 2002 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/0020772021000046216

(1988, 1990a, b), which are easy to train, so the training process can be implemented online. Using this implementation to estimate states of nonlinear systems, the obvious advantages are that the accuracy of the estimations and the tracking ability can be determined by design.

In this paper, after an introduction of a probability neural network filter for nonlinear system, a more reasonable formula for the fault threshold is derived based on the properties of PNNF. When there is no strong disturbance, the threshold is calculated with the statistical properties of the system noise and the estimation error of the filter. When a disturbance causes the state of the system to change abruptly, the threshold is calculated not only based on the statistical properties of the system noise and the estimation error, but also based on the tracking process corresponding to the abrupt state change. Furthermore, for some special cases when a fault is potential but the system is in steady state, the information for fault detection may be insufficient, and a group of disturbances with a definite amplitude (positive or negative) are input into the system so that the result of detection can be enhanced by comparing the real tracking process of the filter with the anticipated tracking process corresponding to the amplitude of the disturbances.

In Section 2, probabilistic neural networks are briefly introduced, and the training of PNN for nonlinear system filtering and the filtering process are described. In Section 3, the formula for calculating the fault threshold is studied, which is mainly dependent on three factors including the statistical properties of the noise, the estimation error of the filter, and the tracking description of the filter. Also, a fault detection method with artificial disturbances is introduced to deal with the detection of a potential fault when the system is in steady state. In Section 4, two numerical simulations are employed to demonstrate the method of fault detection based on probabilistic neural networks estimation. Finally, Section 5 gives the summary and conclusions.

2. Probabilistic Neural Network Filtering (PNNF)

Since a probabilistic neural network has not been used as widely as other kinds of neural networks such as MLP, a brief introduction to PNN is necessary.

2.1. *Brief description of PNN*

The PNN proposed by Specht (1988, 1990a, b) is a four-layer feed-forward network. The block diagram of a PNN is showed in figure 1. The input layer supplies the same input values to all of the pattern units in the pattern layer. Each pattern unit forms a dot product of the input vector X, with a pattern P_i, $Y_i = X \cdot P_i$, and

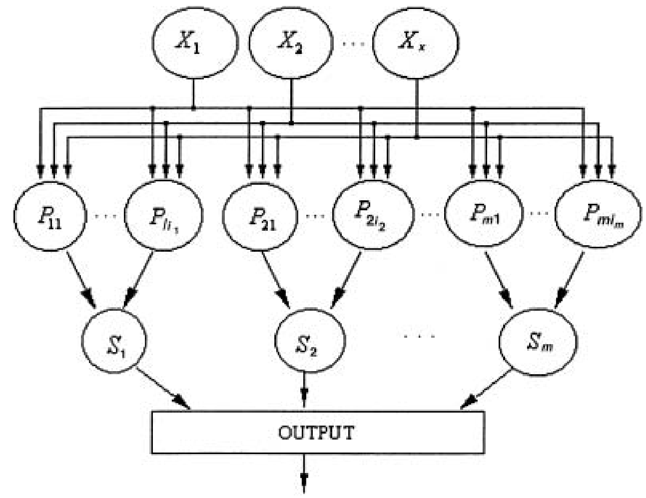

Figure 1. Block diagram of probabilistic neural networks.

then performs nonlinear operations on Y_i before outputting its activation level to the summation layer. The nonlinear operations, for example, can be $\exp[(Y_i - 1)/\sigma^2]$, where σ is the smoothing factor, which is related to the accuracy. Equivalently, if X and P_i are normal vectors, the operations are $\exp[-(P_i - X)^{\mathrm{T}}(P_i - X)/\sigma^2]$. The summation layer simply sums the category from which the training pattern was selected. The output of the summation layer is the estimation of posterior probabilities corresponding to the category. For the application of classification, the output layer can synthesize the *a priori* probabilities and the risk coefficients to make a Bayes optimal decision.

2.2. *Training of PNN for nonlinear system filtering*

The training of a PNN is executed to generate the patterns in the pattern layer and to connect each pattern to the appropriate summation unit. For the application of nonlinear filtering, these patterns are a group of predictive outputs of a nonlinear system. A discrete nonlinear observable system is considered as follows:

$$X_k = \Phi(X_k, k-1) + \Gamma(X_k, k-1)W_k \tag{1a}$$

$$Z_k = h(X_k, k) + V_k, \tag{1b}$$

where Φ, and h are nonlinear mappings, Γ is either linear or nonlinear mapping, $X_k \in R^r$ is the state variable, $Z_k \in R^l$ is the output variable, and $W_k \in R^m$ and $V_k \in R^l$ are Gaussian white noises. These conditions are in the same situation as an EKF. However, here the maps Φ, h and Γ can be substituted by any approximations such as backwards propagation neural networks. There are two statistical variables in system (1), W_k and V_k, which are closely related to the state estimation, and the distribution density functions of $\{W_k\}$ and $\{V_k\}$ are obtainable.

Let $\Lambda(k)$ be the distributed density function of W_k in $[b_{k1}, b_{k2}]$, where:

$$b_{k1} = \inf\{W_i; i = k-N, k-N+1, \ldots, k\} \quad (2a)$$

$$b_{k2} = \sup\{W_i; i = k-N, k-N+1, \ldots, k\} \quad (2b)$$

and n is the length of the sliding data window.

For generating a group of predictive output values of system (1), the interval $[b_{k1}, b_{k2}]$ is first divided and a group of values of the division points obtained. Then, the statistical variable W_k in equation (1a) is substituted with these values of the division points and a group of candidate values of state estimation can be calculated. Using these candidate values of state estimation, a group of predictive output values can be generated by equation (1b), which are regarded as the patterns and stored in the pattern layer of PNN.

The estimation error and tracking ability of the filter are mainly dependent on how the interval $[b_{k1}, b_{k2}]$ is divided. Let η be the maximum permissible error of the state estimation. For η, we can find a δ, such that:

$$|\Gamma(X_{k-1}, k-1)\delta| < \eta. \quad (3)$$

Expression (3) aims that the noise interval $[b_{k1}, b_{k2}]$ is divided fine enough (not exceed to δ) so that the expected estimation accuracy η can be reached.

For estimating the state with expected accuracy and tracking efficiency corresponding to strong disturbances, the disturbance interval $[b_{k1}, b_{k2}]$ needs to be divided internally and externally, and the steps A, B and C are as follows:

Step A. Interval $[b_{k1}, b_{k2}]$ is divided internally on the basis of the function $\Lambda(k)$, which represents the probabilistic density distribution of the interval. The points are $L_i; i = 0, 1, \ldots, j$, where $L_0 = b_{k1}$, $L_j = b_{k2}$, and $\max(|L_{i+1} - L_i|) \le \delta$; $i = 0, 1, \ldots, j-1$. This means that it is equally probable that W_k belongs to any of the sub-intervals $[L_i, L_{i+1}); i = 0, 1, \ldots, j-1$. This is especially true if the probabilistic density is homogeneously distributed and the lengths of the intervals above are equal.

If disturbances of the system are always in the interval $[b_{k1}, b_{k2}]$, this step can ensure the PNN output to approximate a Bayes optimal result. But in practice, sometimes there are strong disturbances that cannot be limited in the interval $[b_{k1}, b_{k2}]$. Therefore, steps B and C are introduced.

Step B. Two intervals adjacent to $[b_{k1}, b_{k2}]$ are divided, namely the intervals $[\frac{1}{2}(3b_{k1} - b_{k2}), b_{k1})$ and $(b_{k2}, \frac{1}{2}(3b_{k2} - b_{k1})]$. An integer p can be determined such that the division points for the intervals $[\frac{1}{2}(3b_{k1} - b_{k2}), b_{k1})$ and $(b_{k2}, \frac{1}{2}(3b_{k2} - b_{k1})]$ satisfy:

$$\begin{aligned} b_{k1} - L_{-1} &= L_{-1} - L_{-2} \\ &= \cdots = L_{-p+1} - \tfrac{1}{2}(3b_{k1} - b_{k2}) \le \delta \end{aligned} \quad (4a)$$

$$L_{-p} = \tfrac{1}{2}(3b_{k1} - b_{k2}) \quad (4b)$$

$$\begin{aligned} b_{k2} - L_1 &= L_1 - L_2 \\ &= \cdots = L_{p-1} - \tfrac{1}{2}(3b_{k2} - b_{k1}) \ge -\delta \end{aligned} \quad (4c)$$

$$L_p = \tfrac{1}{2}(3b_{k2} - b_{k1}). \quad (4d)$$

This step is aimed at moving the estimation points into the interval $[b_{k1}, b_{k2}]$ from outside during the tracking process.

Step C. Outside the interval

$$[\tfrac{1}{2}(3b_{k1} - b_{k2}), \tfrac{1}{2}(3b_{k2} - b_{k1})],$$

the division points are defined as:

$$L_{-p-1} = L_{-p} - 2^0(b_{k2} - b_{k1}) \quad (5a)$$

$$L_{-p-2} = L_{-p-1} - 2^1(b_{k2} - b_{k1}) \quad (5b)$$

$$\ldots$$

$$L_{-p-M} = L_{-p-M+1} - 2^{M-1}(b_{k2} - b_{k1}) \quad (5c)$$

$$L_{j+p+1} = L_{j+p} + 2^0(b_{k2} - b_{k1}) \quad (5d)$$

$$L_{j+p+2} = L_{j+p+1} + 2^1(b_{k2} - b_{k1}) \quad (5e)$$

$$\ldots$$

$$L_{j+p+M} = L_{j+p+M-1} + 2^{M-1}(b_{k2} - b_{k1}) \quad (5f)$$

The points

$$L_i; \quad i = -p-M, -p-M+1, \ldots, -p, j+p, j+p+1, \ldots, j+p+M$$

are tracking points. The structure of the tracking points described above is just an example, other structures can be chosen dependent on the application.

Up to this step, all of the values of division points of interval $[b_{k1}, b_{k2}]$ both internal (for state estimation) and external (for tracking) have been obtained.

System (1) is supposed to be observable and stable. Substituting W_k in (1a) with

$$L_i; \quad i = -p-M, -p-M+1, \ldots, j+p+M-1, j+p+M,$$

a group of candidate states are generated by (1a), which are

$$X_i^*; \quad i = -p - M, -p - M + 1, \ldots, j + p + M - 1, j + p + M.$$

Consequently, a group of predictive output values can be generated by (1b) as

$$\hat{Z}_i; \quad i = -p - M, -p - M + 1, \ldots, j + p + M - 1, j + p + M,$$

which are regarded as the patterns and kept in the pattern layer of the PNN.

Up to this step, the training of the PNN is finished and it is ready for the arrival of an input vector.

2.3. *PNN filtering process*

Since the training of the PNN has finished, a group of predictive output values of system (1) are stored in the pattern layer of PNN. The output vector of system (1) will be regarded as the input of PNN, and the process of the state estimation of system (1) is described as follows:

Step A. Input layer: The input layer of the PNN transmits the input vector Z_k (the output vector of system (1) to each node of the pattern layer.

Step B. Pattern layer: In the pattern layer, there are $2(p + M) + j + 1$ patterns. After receiving the input vector, this layer generates a group of predictive disturbance values as

$$\{|(\hat{Z}_i - Z_k)| : i = -p - M, -p - M + 1, \ldots, j + p + M - 1, j + p + M\},$$

where $\{|(\hat{Z}_i - Z_k)|; i = 0, 1, \ldots, j - 1, j\}$ are the predictive values of the measurement noises, and the set

$$\{|(\hat{Z}_i - Z_k)|;\ i = -p - M, -p - M + 1, \ldots, -1, j + 1, \ldots, j + p + M - 1, j + p + M\}$$

represents the differences between the output of the system and the predictive output corresponding to the strong disturbances.

Step C. Summation layer: In the summation layer, the probability values

$$\{P(|\hat{Z}_i - Z_k|; \quad i = -p - M, -p - M + 1, \ldots, j + p + M - 1, j + p + M\}$$

are calculated using the Bayes probability formulation.

Step D. Output layer: In the output layer, a suitable integral

$$i'; \quad i' \in \{-p - M, -p - M + 1, \ldots, j + p + M - 1, j + p + M\}$$

is determined such that:

$$P(|\hat{Z}_{i'} - Z_k|) = \max\{P(|\hat{Z}_i - Z_k|)\}, \tag{6}$$

where

$$i = -p - M, -p - M + 1, \ldots, j + p + M - 1, j + p + M.$$

The output vector of the PNN is $X_{i'}^*$. If $i' \in \{0, 1, \ldots, j - 1, j\}$, then $X_{i'}^*$ will be the closest value to Bayes optimal result of state estimation. If $i' \in \{-p, -p + 1, \ldots, -1, j + 1, \ldots, j + p\}$, then the filter will approximate the Bayes optimal estimation results in the next step. If

$$i' \in \{-p - M, -p - M + 1, \ldots, -p - 1, j + p + 1, \ldots, j + p + M\},$$

then the filter is in the tracking process corresponding to the abrupt state changing.

Then let $\hat{X}_k$ be $X_{i'}^*$, and $\hat{\omega}_k = L_{i'}$ as residual, the training process of PNN can be restarted and the estimation process for $\hat{X}_{k+1}$ is performed.

The block diagram of the PNN filter is shown in figure 2, and fault detection is also included.

Based on the structure of the tracking points described in (5), it is not difficult to derive that during the tracking process, the error transmission rate is

$$\frac{2^{i'}\Gamma(\hat{X}_{k-1}, k - 1) + \Gamma(\hat{X}_k, k)}{3 \cdot 2^{i'}\Gamma(\hat{X}_{k-1}, k - 1)},$$

when

$$\frac{W_k - L_{i'}}{W_k - b_{k1}} < 0,$$

and

$$\frac{W_k - L_{i'}}{W_k - b_{k2}} < 0,$$

and

$$\frac{2^{i'-1}\Gamma(\hat{X}_{k-1}, k - 1) + \Gamma(\hat{X}_k, k)}{2^{i'+1}\Gamma(\hat{X}_{k-1}, k - 1)},$$

when

$$\frac{W_k - L_{i'}}{W_k - b_{k1}} > 0$$

and

$$\frac{W_k - L_{i'}}{W_k - b_{k2}} > 0,$$

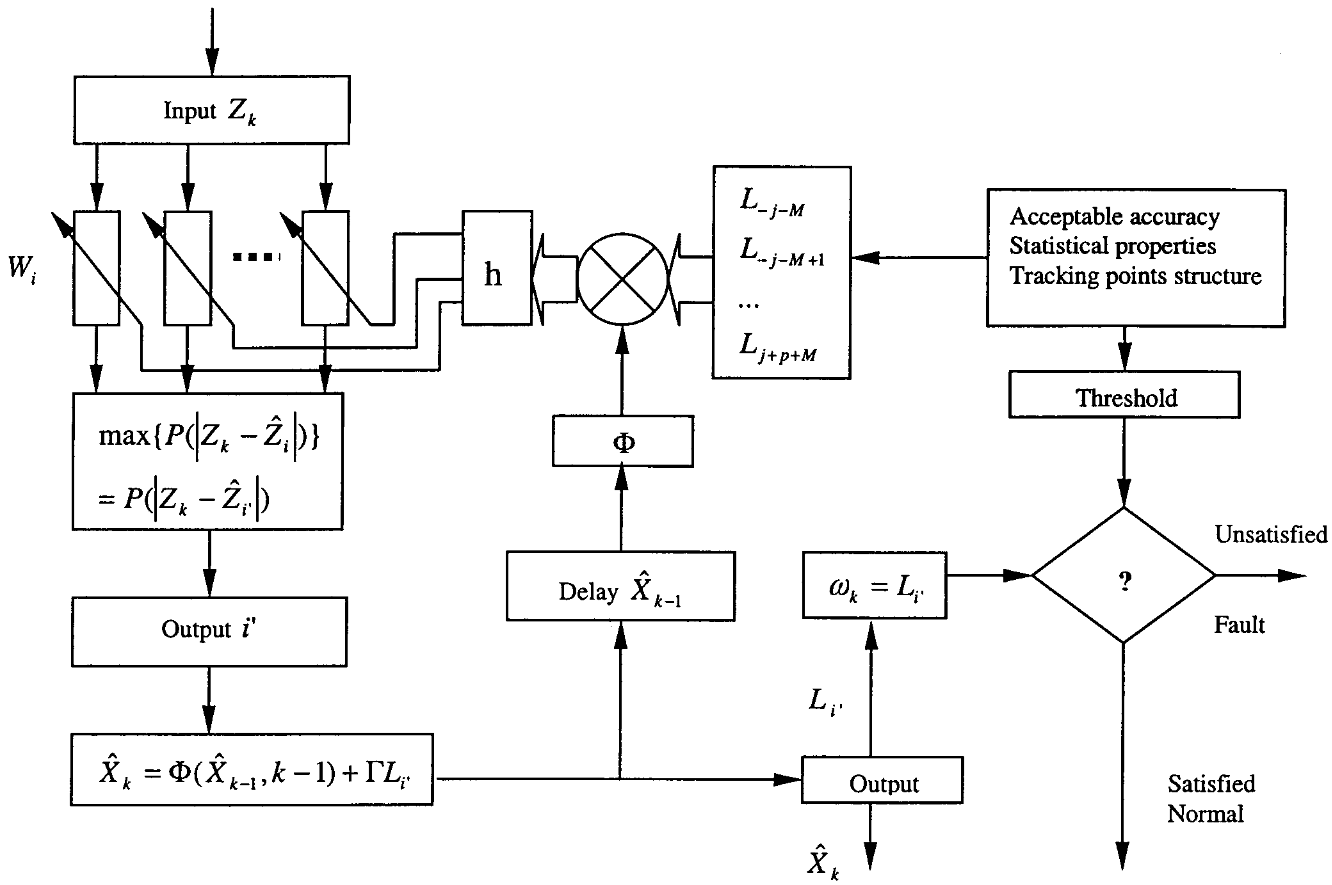

Figure 2. Block diagram of a probabilistic neural network filter and fault detection where '?' means if the inequalities (7), or (10) and (11) are satisfied.

and the maximum estimating error is less than $2^{i'-1}(b_{k2} - b_{k1})$ and $2^{i'}(b_{k2} - b_{k1})$, respectively, where i' describe the location of $L_{i'}$. Therefore the tracking steps corresponding to a strong disturbance can easily be calculated.

3. Fault detection and diagnosis

Using PNNF, the maximum estimation error to the asymptotic Bayes optimal result can be bounded to a given acceptable value when a strong disturbance is involved, and the tracking process corresponding to a strong disturbance which causes the states of the system to change abruptly can be described by an expression. A fault threshold consequently can be calculated based on these properties.

3.1. *Fault threshold*

Let $\bar{W}$ be the average variance of $\{W_k\}$. When the model matches the system and no strong disturbance affects the system to cause an abrupt state changing, the mean variance of the residuals $\{\hat{\omega}_k\}$ should be bounded by $\rho = \bar{W} + 2\eta\sqrt{\bar{W}} + \eta^2$, i.e., the following inequality should be satisfied:

$$\frac{1}{M}\sum_{i_1=k-M}^{k} \hat{\omega}_{i_1}^2 \leq \rho, \tag{7}$$

where N is the length of the sliding data window. Otherwise, some faults must have occurred. When the system is disturbed strongly and changes the state abruptly, the filter will track the changing state. When the model matches the system, the number of the tracking steps corresponding to a strong disturbance can be expressed by a function, which is determined by the structure of the tracking points. For the structure of the tracking point described in (5), the function is:

$$T_r(W_k) = i' - j - p(i' > 0) \tag{8a}$$

$$T_r(W_k) = -i' - p(i' < 0), \tag{8b}$$

where $L_{i'}$ is the estimation value of W_k. The maximum anticipated accumulated estimation errors during the tracking can also be expressed as:

$$\rho_1 = \sum_{i=i_0}^{Tr(W_{i_0})} 2^{i-i_0}(b_{k2} - b_{k1}), \tag{9}$$

where i_0 is the time location when a strong disturbance takes place. Therefore, the following inequality should be satisfied when the model matches the system:

$$\sum_{i=i_0}^{Tr(W_{i_0})} \hat{\omega}_i \le \rho_1. \tag{10}$$

In the situation where the system is disturbed strongly, (7) needs to be modified because the sum of the estimation errors during tracking should be removed from (7). Therefore, (7) is changed to:

$$\frac{1}{N - Tr(W_{i_0})} \sum_{i \in I^*} \hat{\omega}_i^2 \le \rho, \tag{11}$$

where I^* is the sliding data window without the interval corresponding to the tracking process.

Consequently, when the model matches the system and there is a strong disturbance which causes the states of the system to change abruptly, inequalities (10) and (11) should be satisfied synchronally. Otherwise some faults must have occurred. See figure 2 where the block diagram for fault detection is provided.

3.2. *Artificial disturbance*

The method described above can be effective when the system is dynamic. When the model matches the system, the difference between the two trajectories of the state of the system and the filter is bounded by (7). If the error succeeds the limit as (7) described, some faults must have occurred. However, sometimes when the system is in steady-state, there could be a potential fault which cannot be clearly detected because it occurred when the system had the same static state as the fault-free system. To deal with this situation, in some special cases the systems can load some artificial disturbances that cause the state to change abruptly so that the system is in a dynamic situation temporarily. Simultaneously with the artificial disturbances, the PNNF will track the change of the state and the tracking process can also be anticipated. Therefore, whether or not there is a fault in the system can be judged by comparing the anticipated response with the actual response of the filter.

Let D be the amplitude of the artificial disturbance W^*_{k-1}, which causes the system to be temporarily dynamic, and $e(D)$ be the sum of the maximum estimation error during the tracking process, which can be calculated as $e(D) = \sum_{i=1}^{Tr(D)} 2^{i-1}(b_{k2} - b_{k1})$, where $Tr(D)$ represents the number of the tracking steps. Let β be the interval of each artificial disturbance, the input of the disturbances is as follows:

(1) A group of artificial disturbances with the same interval and amplitude are input to the system, so (1a) becomes:

$$X^*_k = \Phi(X^*_{k-1}, k-1) + \Gamma(X^*_{k-1}, k-1)(W_k + D\delta_{(k-1)(i\beta)}), \tag{11}$$

where $D > 0, 0 < \beta < T_r(D), i = 0, 1, 2, \ldots, i_\beta$.

Let N_1 be the length of the sliding data window, and N_1 can be suitably determined so that

$$I_\beta = \frac{N_1}{\beta} = \left[\frac{N_1}{\beta}\right], \qquad \text{where} \qquad \left[\frac{N_1}{\beta}\right]$$

expresses the maximum integer not greater than N_1/β. The fault can be detected by checking if the following inequalities are satisfied:

$$\sum_{i=i_1}^{i_1+T_r(D)} |\hat{\omega}_i| \le e(D) \tag{12a}$$

$$\sum_{i=i_2}^{i_2+T_r(D)} |\hat{\omega}_i| \le e(D) \tag{12b}$$

$$\cdots$$

$$\sum_{i=i_\beta}^{i_\beta+T_r(D)} |\hat{\omega}_i| \le e(D) \tag{12c}$$

$$\sum_{i=k-N_1}^{k} |\hat{\omega}_i| \le I_\beta e(D) + (N_1 - I_\beta \cdot Tr(D))\rho. \tag{13}$$

The inequalities (12) are for evaluating each tracking process and inequality (13) summarizes all of the tracking processes, the estimating error and the statistical properties of the noise.

(2) A group of artificial disturbances with the same intervals and alternative reverse amplitude are inputs to the system, so (1a) becomes:

$$X^*_k = \Phi(X^*_{k-1}, k-1) + \Gamma(X^*_{k-1}, k-1)(W_k + (-1)^{k-1} D\delta_{(k-1)(i\beta)}). \tag{14}$$

When some parameters of the system are changed, the sensitivity of the system to the positive amplitude disturbance and the negative amplitude disturbance may be different. Therefore this artificial disturbance pattern can be more effective for fault detection, as well as providing more information for the diagnosis of the fault when $|e(D)|$ is compared with $|e(-D)|$.

(3) More complex patterns of the artificial disturbances can be organized to enter the system corresponding to different fault patterns, and a set of fuzzy clusters can thus be generated which can be used for fault diagnosis. Since this topic is not the main purpose of this paper, no details are discussed here.

4. Simulation

Two simulations are employed here to demonstrate the method described above.

The two numerical examples are local observable and stable nonlinear systems, which mainly include nonlinear and linear terms. The nonlinear terms are with small coefficients but high nonlinearities. For each pair of the initial values that differ from the steady-states of the systems, the trajectories are different, so one can demonstrate the tracking ability of the filter using a PNN.

In the first simulation, the initial values of the filter are significantly different from the initial values of the system. The system is detected for faults in four situations: fault free without a strong disturbance, fault free with strong disturbances, fault occurred without a strong disturbance and fault occurred with strong disturbances.

In the second simulation, the initial values of both the system and the filter are same. Three situations are discussed: fault free with artificial disturbance, fault occurred but failed to be clearly detected without artificial disturbance and fault occurred with artificial disturbance.

4.1. *Simulation 1*

Consider a nonlinear system as follows:

$$x_1(k) = 0.1[0.7 + x_2(k-1)]\cos[0.7 + x_2(k-1)] + 0.9x_2(k-1) \tag{15a}$$

$$x_2(k) = 0.1[0.7 + x_2(k-1)]\sin[x_1(k-1) + x_2(k-1)] + Ax_2(k-1) + 0.43x_1(k-1) + \omega_{k-1} \tag{15b}$$

$$Z(k) = x_2^2(k) - 63.5x_2(k) + v_k, \tag{15c}$$

where $\{\omega_k\}$ and $\{v_k\}$ are mean distributed white noises, $A = 0.51$, and $\|Q_k\| \leq 0.25$ and $\|R_k\| \leq 0.25$.

To implement the filter, the maps in the state equations (15a, b) are approximated using MLP, where $x_1^* \in (-24, 24)$ and $x_2^* \in (-24, 24)$. The training of the MLP is performed off-line and the level of the error of the one-step prediction is limited to 10^{-4}.

The filter is as follows:

$$\begin{pmatrix} x_1^*(k) \\ x_2^*(k) \end{pmatrix} = F_{\mathrm{MLP}}[x_1^*(k-1), x_2^*(k-1)] + \begin{pmatrix} 0 \\ L_i \end{pmatrix} \tag{16a}$$

$$Z^*(k) = x_2^{*2}(k) - 63.5x_2^*(k) + v_d. \tag{16b}$$

The interval [–1, 1] is divided into 40 subintervals, that means $j + p = 20, M = 10$, and the tracking points are $-L_{-11} = 2 = L_{31},\ \ -L_{-12} = 4 = L_{32}, \ldots, -L_{-20} = 20 = L_{40}$. When the disturbance is in the interval $[–21, 21]$ and the state changes abruptly, the filter can catch up in no more than two steps. The estimation accuracy is limited to 0.025. Having synthesized the statistical characteristics of the noise, the estimation accuracy limitation and the estimation error during each tracking process, the threshold is determined as 0.275.

The initial values of system (15) are $x_1(0) = 0.0$, $x_2(0) = 0.0$, and the initial values of the filter (16) are significantly different, being $x_1^* = 8.0$, $x_2^* = 8.6$.

For the figures in this section, the top with the dotted line is the state variable X_2 of the system, and the solid line is the state variable estimation value X_2^* at each time. The middle part represents the noise and artificial disturbance at the relevant times. The bottom is the fault threshold and the detection results.

- Filter (16) is very close to system (15) without strong disturbance. Figure 3 indicates that there is no fault in the system.
- Filter (16) is very close to system (15). In contrast to the first point above, strong disturbances occur at $k = 103$, 201, 249, 360, 429, 540, and the amplitudes are individually 10.0, –10.0, 8.9, 8.7, –6.9, –7.2. Figure 4 indicates that there is no fault in the system.
- In system (15), the parameter A, however, changes to 0.65, and there is no strong disturbance involved. Figure 5 indicates that there is fault in the system.
- In system (15), the parameter A changes to 0.44. Similar to the second point above, there are disturbances at $k = 103$, 201, 249, 360, 429, 540, and the amplitudes are individually 10.0, –10.0, 8.9, 8.7, –6.9, –7.2. Figure 6 indicates that there is fault in the system.

4.2. *Simulation 2*

Consider a nonlinear system as follows:

$$x_1(k) = 0.1x_2(k-1)\sin x_2^2(k-1) + 0.75x_2(k-1) \tag{17a}$$

$$x_2(k) = 0.1x_2(k-1)\cos[x_1(k-1) + x_2(k-1)]$$

$$Bx_2(k-1) + 0.23x_1(k-1) + w \tag{17b}$$

$$Z(k) = 0.5x_2^2(k) - 0.53x_2(k) + v, \tag{17c}$$

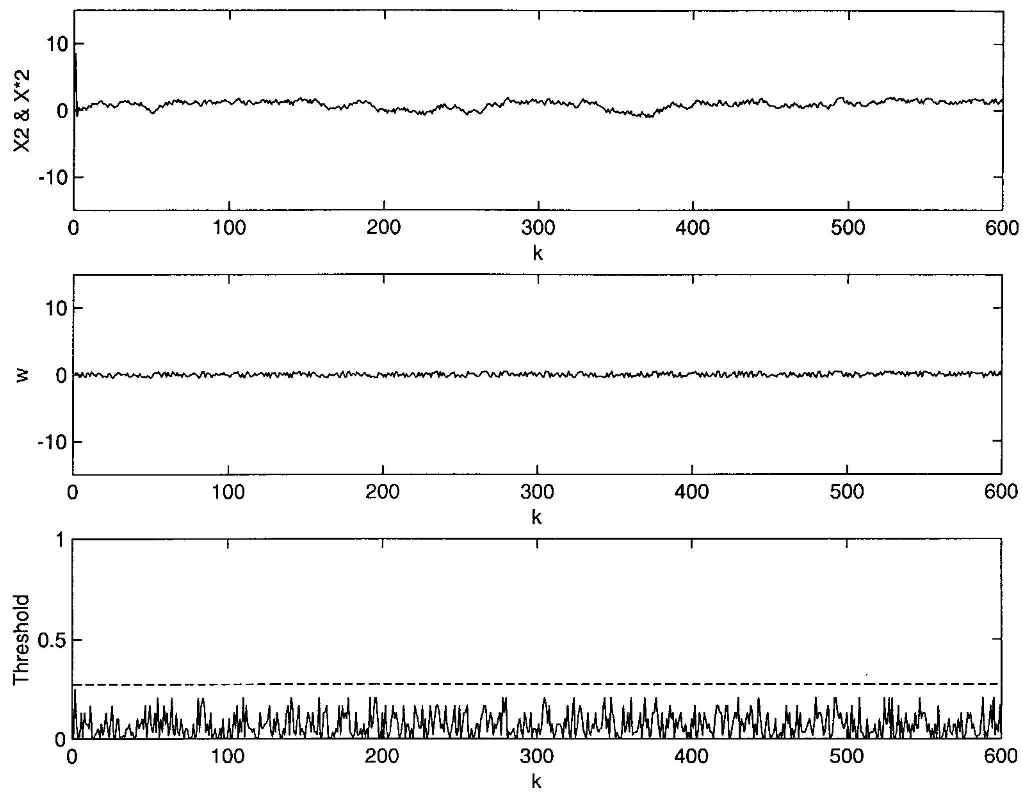

Figure 3. Result of detection for a fault-free system without a strong disturbance.

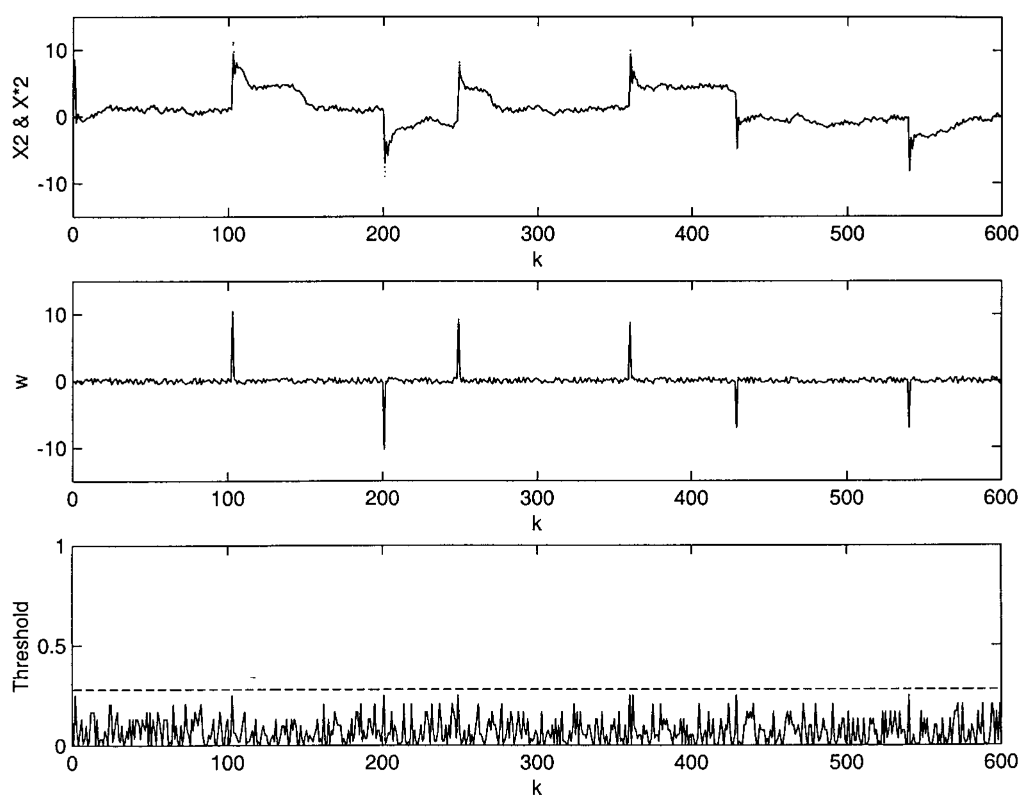

Figure 4. Result of detection for a fault-free system with strong disturbances.

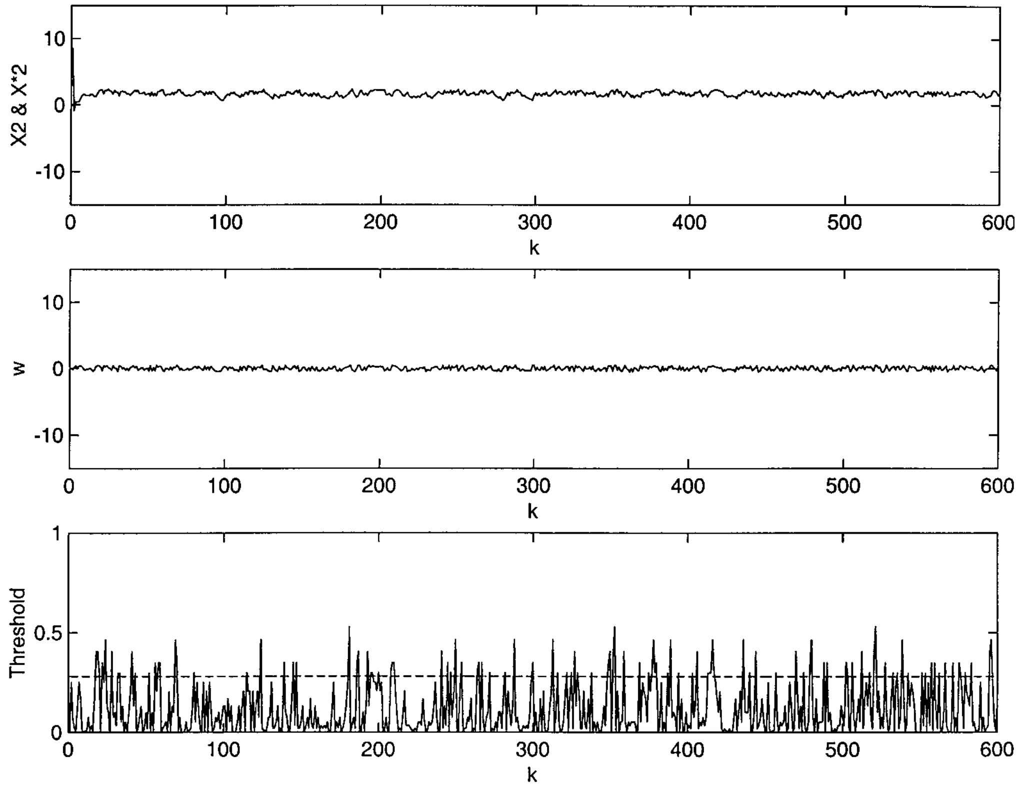

Figure 5. Results of detection for a fault-occurred system without a strong disturbance.

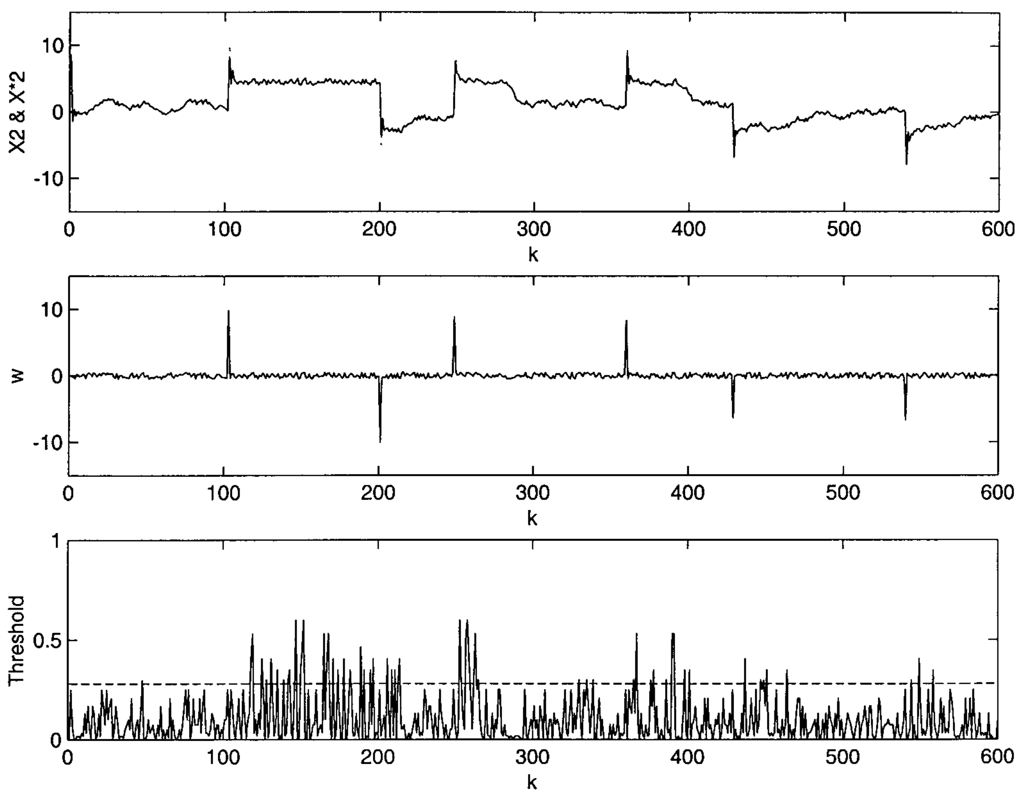

Figure 6. Result of detection for a fault occurred system with strong disturbances.

where w and v are mean distributed white noises, and $B = 0.427$.

Similar to the filter in simulation 1, the maps in the states equations (17a, b) are approximated using MLP where $x_1^* \in (-19, 19)$ and $x_2^* \in (-19, 19)$. The training of the MLP is performed off-line and the level of the error of the one-step prediction is limited to 10^{-4}. The filter is as follows:

$$\begin{pmatrix} x_1^*(k) \\ x_2^*(k) \end{pmatrix} = F_{\mathrm{MLP}}[x_1^*(k-1), x_2^*(k-1)] + \begin{pmatrix} 0 \\ L_i \end{pmatrix} \quad (18a)$$

$$Z^*(k) = 0.5x_2^{*2}(k) - 0.53x_2^*(k) + v. \quad (18b)$$

The structure of the division points, the estimation accuracy and the threshold are the same as those in simulation 1. The initial values of the system are $x_1(0) = x_2(0) = 0$, and the initial values of the filter are $x_1^*(0) = x_2^*(0) = 0$.

- Filter (18) is very close to system (17). The artificial disturbance, with an amplitude of 11.4, is input to the system at $k = 100$, 200, 300, 400 and 500. The artificial disturbance, with an amplitude of -11.4, is input to the system at $k = 150$, 250, 350, 450, and 550. Figure 7 shows that there is no fault in system (17). Equally, the variance of the disturbance estimating series is under the threshold.
- Parameter B becomes 0.54, but the state variable x_2 is still kept around 0 as in the above point. Here the artificial disturbance is not employed. The variance of the disturbance estimating series is only slightly above the threshold, the fault, however, has not been detected clearly, as figure 8 shows.
- Artificial disturbances, exactly as in the first point above, are input into the system (17) when B becomes 0.54. The result of detection indicates that there is fault in system (17), also, the fault is sensitive to the strong disturbance with negative amplitude, as figure 9 shows.

5. Conclusion

The PNN filter has the properties that can limit the estimation error of the asymptotic Bayes optimal result, and the tracking process of PNNF can also be limited by a certain tracking process which responds to the amplitude of an abrupt disturbance. Based on these properties, when no strong disturbance is involved, the threshold is calculated with the statistics of the noise and the estimation errors. When the states change abruptly caused by a strong disturbance, the threshold is determined by the statistics of the noise, and the estimation errors and the expected tracking process which is related

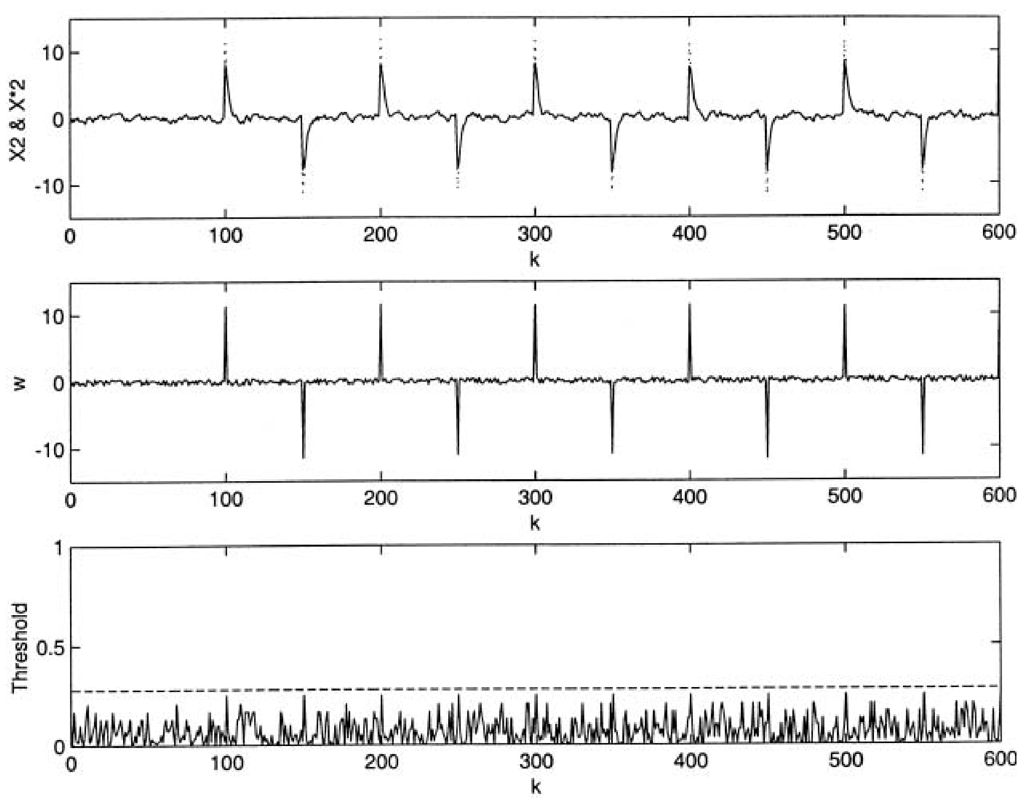

Figure 7. Result of detection for a fault free system with artificial disturbances.

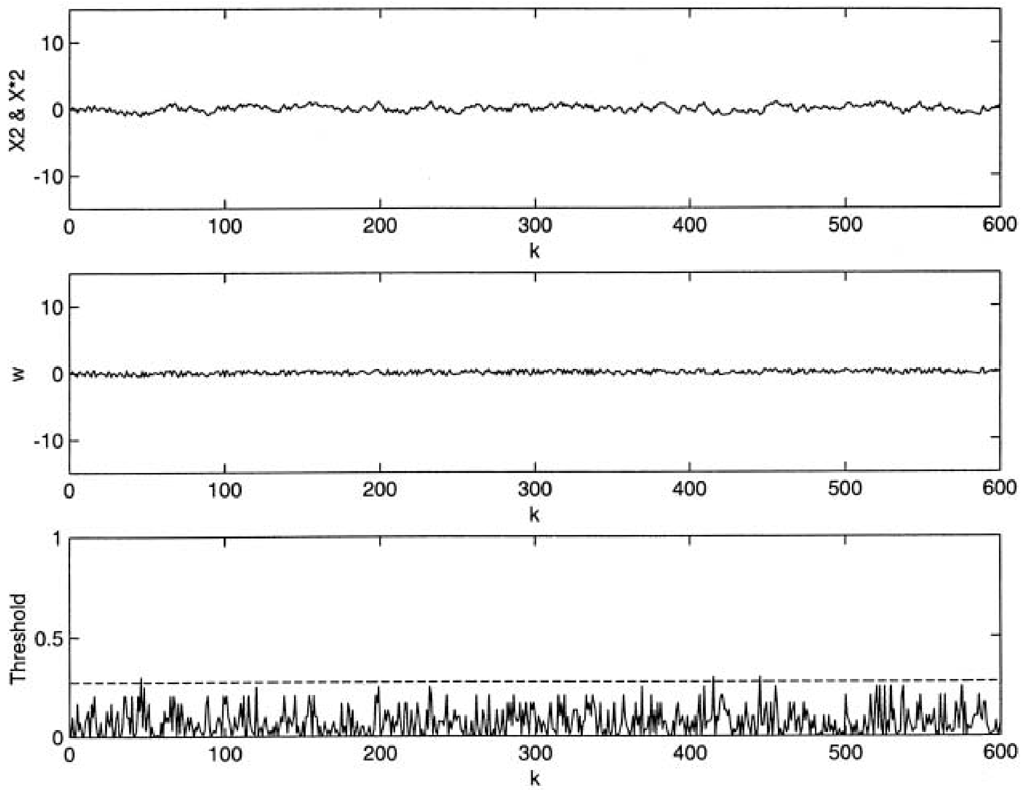

Figure 8. Result of detection for a fault occurred system without artificial disturbances.

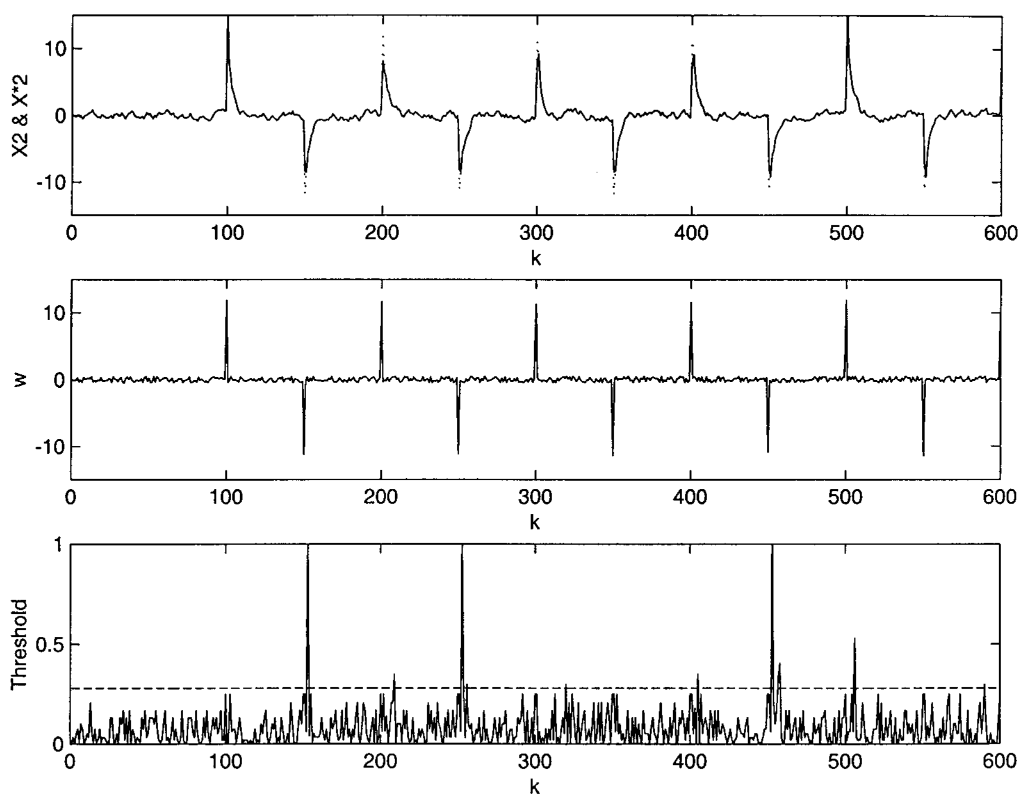

Figure 9. Result of detection for a fault occurred system with artificial disturbances.

to the amplitude of the disturbance. Also, when the parameters of the systems are changed but not their steady-states, the result of detection can be enhanced by inputting a disturbance artificially and comparing the actual tracking process with the expected process.

The method that inputs an artificial disturbance can be improved further by generating a set of fault patterns for the systems (parameters changing). Furthermore, future research involves the generation of a set of fuzzy clusters from which more information about the fault diagnosis can be obtained.

References

CHOW, E. Y., and WILLSKY, A. S., 1984, Analytical redundancy and the design of robust failure detection filters. *IEEE Transactions on Automatic Control,* **29,** 603–614.

DAVS, M. H. A., 1974, The application of nonlinear filtering to fault detection in linear systems. *IEEE Transactions on Automatic Control,* **19,** 257–259.

FRANK, P. M., 1987, Fault diagnosis in dynamic systems via state estimation-A survey. In *Proceedings of the First European Workshop on Fault Diagnostics, Reliablity and Related Knowledge Based Approaches,* (Dordrecht: Reidel), pp. 35–98.

FRANK, P. M., and DING, X., 1994a, Frequency domain approach to optimally robust residual generation and evaluation for model-based fault diagnosis. *Automatica,* **30,** 789–804.

FRANK, P. M., 1994b, On-line fault detection in uncertain nonlinear system using diagnostic observers: a survey. *International Journal of Systems Science,* **25,** 2129–2154.

GUAN, Y., and SAIF, M., 1990, Novel approach to the design of unknown input observers for single output nonlinear systems. *IEEE Transactions on Automatic Control,* **35,** 1054–1058.

ISERMANN, R., 1984, Process fault detection based on modeling and estimation method—a survey. *Automatica,* **20,** 387–404.

KALMAN, R. E., and BUCY, S., 1961, New results in linear filtering and prediction theory. *Transaction of the ASME, Series D: Journal of Basic Engineering,* **83,** 95–108.

SCHMIDT, S. F., 1970, *Computational Techniques in Kalman Filtering, in Theory and Application of Kalman Filtering*. (London: AGARD).

SPECHT, D. F., 1988, Probabilistic neural networks for classification, mapping or associative memory. *Proceedings of the IEEE 1st International Conference on Neural Network,* San Diego, CA, 525–532.

SPECHT, D. F., 1990a, Probabilistic neural networks and the polynomial adaline as complementary techniques for classification. *IEEE Transactions on Neural Networks,* **1,** 111–121.

SPECHT, D. F., 1990b, Probabilistic neural networks. *Neural Networks,* **3,** 109–118.

TRUNOV, A. B., and POLYCARPOU, M. M., 2000, Automated fault diagnosis in nonlinear multivariable systems using a learning methodology, *IEEE Transactions on Neural Networks,* **11,** 91–100.

VEMURI, A. T., and POLYCARPOU, M. M., 1997, Neural-network-based robust fault diagnosis in robotic systems. *IEEE Transactions on Neural Networks,* **8,** 1411–1419.

WHITE, J. E., and SPEYER, J. L., 1987, Detection filter design: spectral theory and algorithms. *IEEE Transactions on Automatic Control,* **32,** 593–603.

WILLSKY, A. S., 1976, A survey of design methods for failure detection in dynamic systems. *Automatica,* **12,** 601–611.

YU, D.,, and SHIELDS, D. N., 1996, A bilinear fault detection observer. *Automatica,* **32,** 1597–1602.

ZHANG, Q., BASSEVILLE, M., and BENVENISTE, A., 1998, Fault detection and isolation in nonlinear dynamic systems: a combined input-output and local approach. *Automatica,* **34,** 1359–1373.